Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Please amend the claims as follows:

1. (Currently Amended) A method for tracing program flow within an application server comprising:

performing the following before loading classfiles of application components for processing at runtime:

identifying ~~one or more of~~ the application components to be traced within the application server;

modifying bytecode associated with the ~~one or more~~ identified application components, the modifications associated with a particular set of methods of the application components related to program execution across application servers, databases and/or external systems;

performing the following at runtime:

loading the respective classfiles of the ~~one or more~~ identified application components, the loading including identifying the names and methods of the respective classfiles with a dispatcher;

executing the ~~one or more~~ identified application components, the executing including dispatching method invocations to ~~a~~ respective plug-ins;

with the plug-ins, registering the method invocations and method-related information associated with the particular set of methods; and

translating the method-related information to a format employed within a distributed statistical records ("DSR") system and forwarding the translated information to the DSR system.

2. (Original) The method as in claim 1 wherein one the application components are Java application components.

3. (Currently Amended) The method as in claim 1 wherein the application servers [[is a]] are Java 2 Enterprise Edition ("J2EE") servers and the application components are J2EE services within the J2EE servers.

4. (Original) The method as in claim 1 further comprising: storing the method-related information within a plurality of DSR files within the DSR system.

5. (Original) The method as in claim 1 wherein modifying the bytecode comprises:

inserting a start method invocation prior to each method of the set of methods and inserting an end method invocation following each method of the set of methods.

6. (Currently Amended) The method as in claim 1 wherein the method-related information comprises an amount of time it takes for ~~each~~ at least one method within the set of methods to complete.

7. (Currently Amended) The method as in claim 1 wherein the method-related information comprises a number times that ~~each~~ at least one method of the set of methods is executed.

8. (Currently Amended) The method as in claim 1 wherein the method-related information comprises input and/or output parameters associated with ~~each~~ at least one method of the set of methods.

9. (Currently Amended) The method as in claim 1 wherein the particular set of methods comprise entry and/or exit ~~methods for each application component, the entry/exit methods representing entry and exit~~ points ~~to and from each component~~.

10. (Currently Amended) The method as in claim 9 wherein the entry/exit ~~methods~~ points are entry and exit points between an application component and an external system.

11. (Currently Amended) The method as in claim 9 wherein the entry/exit ~~method are entry and exit~~ points are between an application component and a database containing data usable by the application

component.

12. (Currently Amended) A system for tracing program flow within an application server comprising:

a distributed statistical records ("DSR") module to identify ~~one or more~~ application components within the application server to be traced;

a bytecode modification module to responsively modify the bytecode of the ~~one or more~~ application components before their respective classfiles are loaded for processing at runtime, the modifications associated with a particular set of methods of the application components related to program execution across application servers, databases and/or external systems;

a dispatch unit to, during runtime, receive the classfile name and method name from each classfile of the respective classfiles as part of its classloading process, and, dispatch to ~~a~~ respective plug-in modules method invocations from objects created from the respective classfiles, the plug-in modules to register method invocations and method-related information associated with the particular set of methods and to provide the method-related information to the DSR module; and

a DSR interface module to translate the method-related information to a format employed within a distributed statistical records ("DSR") system and forward the translated information to the DSR system.

13. (Currently Amended) The system as in claim 12 wherein ~~one~~ the application components are Java application components.

14. (Original) The system as in claim 12 wherein the application server is a Java 2 Enterprise Edition ("J2EE") server and the application components are J2EE services within theJ2EE server.

15. (Original) The system as in claim 12 further comprising: a DSR storage server to store the method-related information within a plurality of DSR files within the DSR system.

16. (Currently Amended) The system as in claim 12 wherein, to modify the bytecode, the bytecode modification module inserts a start method invocation ~~prior~~ proximate to a respective start of each method of the set of methods and inserting an end method invocation ~~following~~ proximate to a respective end of each method of the set of methods.

17. (Currently Amended) The system as in claim 12 wherein the method-related information comprises an amount of time it takes for ~~each~~ at least one method within the set of methods to complete.

18. (Currently Amended) The system as in claim 12 wherein the method-related information comprises a number of times that ~~each~~ at least one method of the set of methods is executed.

19. (Currently Amended) The system as in claim 12 wherein the method-related information comprises input and/or output parameters associated with ~~each~~ at least one method of the set of methods.

20. (Currently Amended) The system as in claim 12 wherein the particular set of methods comprise entry and/or exit ~~methods~~ points for each application component~~, the entry/exit methods representing entry and exit points to and from each component~~.

21. (Currently Amended) The system as in claim 20 wherein the entry/exit ~~methods~~ points are entry and exit points between an application component and an external system.

22. (Currently Amended) The system as in claim 20 wherein the entry/exit ~~method~~ points are entry and exit points between an application component and a database containing data usable by the application component.

23. (Currently Amended) An article of manufacture including program code which, when processed by a machine, causes the machine to perform the operations of:
performing the following before loading classfiles of application components for processing at runtime:

identifying ~~one or more of~~ the application components to be traced within the application server;

modifying bytecode associated with the one or more identified application components, the modifications associated with a particular set of methods of the application components related to program execution across application servers, databases and/or external systems;

performing the following at runtime:

loading the respective classfiles of the ~~one or more~~ identified application components, the loading including identifying the names and methods of the respective classfiles with a dispatcher;

executing the ~~one or more~~ identified application components, the executing including dispatching method invocations to a plug-in;

with the plug-in, registering the method invocations and method-related information associated with the particular set of methods; and

translating the method-related information to a format employed within a distributed statistical records ("DSR") system and forwarding the translated information to the DSR system.

24. (Original) The article of manufacture as in claim 23 wherein one the application components are Java application components.

25. (Currently Amended) The article of manufacture as in claim 23 wherein the application server is a Java 2 Enterprise Edition ("J2EE") server ~~and the application~~.

26. (Original) The article of manufacture as in claim 23 comprising additional program code to cause the machine to perform the operations of:

storing the method-related information within a plurality of DSR files within the DSR system.

27. (Currently Amended) The article of manufacture as in claim 23 wherein modifying the bytecode comprises:

inserting a start method invocation ~~prior to~~ proximate to a respective start of each method of the set of methods and inserting an end method invocation ~~following~~ proximate to a respective end of each method of the set of methods.

28. (Original) The article of manufacture as in claim 23 wherein the method-related information comprises an amount of time it takes for each method within the set of methods to complete.

29. (Currently Amended) The article of manufacture as in claim 23 wherein the method-related information comprises a number of times that each method of the set of methods is executed.

30. (Original) The article of manufacture as in claim 23 wherein the method-related information comprises input and/or output parameters associated with each method of the set of methods.

31. (Currently Amended) The article of manufacture as in claim 23 wherein the particular set of methods comprise entry and/or exit ~~methods~~ points for each application component,
the entry/exit methods representing entry and exit points to and from each component.

32. (Currently Amended) The article of manufacture as in claim 31 wherein the ~~entry/exit methods are~~ entry and exit points are between an application component and an external system.

33. (Currently Amended) The article of manufacture as in claim 31 wherein the ~~entry/exit method are~~ entry and exit points are between an application component and a database containing data usable by the application component.